Evaluation of Health Care System Model Based on Collaborative Algorithms

The final publication is available at link.springer.com

Introduction

The rapid development and use of information and communication technologies in the last two decades has influenced a dramatic transformation of public health and health care, changing the roles of the health care support systems and services. Recent trends in health care support systems are focused on developing patient-centric pervasive environments and the use of mobile devices and technologies in medical monitoring and health care systems [1].

Nowadays, myriad of different sensors are used to monitor human's vital parameters such as: heart rate, blood pressure, blood sugar level. These sensors can be connected to a smartphone. The patient can use the software installed on the smartphone to read the sensor data and the software can present this data directly to the patient or can send it to the doctor which can further analyze it. This is the simplest scenario in mHealth. Smarter system can make conclusions about the current health condition of the patient by using the sensor data and integrated expert knowledge. The system could inform the patient or the doctor when some of the patient's health parameters are not in the normal range.

The bigger challenge in mHealth is to develop a system which could give advices to the patients in order to improve their health condition. Patients could be advised to eat some particular food more often, to perform specific activities, to take some medicine or to visit a doctor for physical examination. If the process of making advices is done entirely by a doctor which has remote access to the mobile application used by the patient, it could be expensive and slow. Better approach could be to automate the process of making advices and recommendations. For this purpose the system will need all data that could be useful: the current health condition, the history of measurements, the patient's diagnosis and an expert's knowledge. Additionally, the types of recommendations and the algorithm which will be used to generate them should also be defined. Although the expert's knowledge will be used, the elimination of the direct human factor from the process of making advices means that the generated advices will be with smaller relevance – additional detailed physical examination could reveal more useful information. It is advised to use the system in a domain where the generated recommendations could potentially improve the health condition of the patient, but they could not worsen it.

In this context, introduction of a novel patient-centric collaborative health care system model COHESY [1], with an integrated intelligent recommendation algorithm, could give a new dimension to preventive and curative medicine. The proposed model helps its users to actively participate in their health care and prevention, thereby providing an active life in accordance with their daily responsibilities at work, family and friends.

COHESY uses mobile, web and broadband technologies, so the citizens have ubiquity of support services where ever they may be, rather than becoming bound to their homes or health centers as pointed out by different authors [2]. Broadband mobile technology provides movements of electronic care environment easily between locations and internet-based storage of data allows moving location of support. Integrated social network in COHESY, allows communication between users with same or similar condition and exchange of their experiences. The social network provides data from all users which can be used by the recommendation algorithm.

The recommendation algorithm in COHESY gives recommendations about physical activities that should be performed by the patients in order to improve their health condition. The algorithm is based on the dependence between the values of the health parameters and the users' physical activities (e.g. walking, running, biking). The basic idea is to find out which physical activities cause change (improvement) of the value of health parameters. This dependence continues to be used by the algorithm to recognize the same or similar health conditions found in another user with similar characteristics. To achieve this, in the proposed recommendation algorithm classification and filtering algorithms are applied in order to group users with similar characteristics. The usage of classified data provides relevant recommendations based on prior knowledge of users with similar health conditions and reference parameters. This way, COHESY bridges the gap between users, clinical staff and medical facilities, strengthening the trust between them and providing relevant data from a larger group of users, grouped on the basis of various indicators.

1. COHESY overview

COHESY is deployed over three basic usage layers (Fig.1). The first layer consists of the bionetwork (implemented from various body sensors) and a mobile application that reads sensor data and allows the users to specify the activities they perform. The application collects users' bio data during various physical activities (e.g. walking, running, cycling). Additionally, the application collects other sensor data (e.g. GPS coordinates) that can be used to describe the performed activities (e.g. speed, duration).

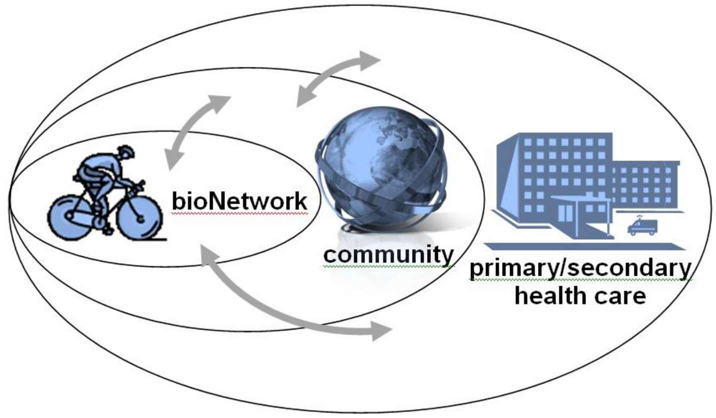

Figure 1. Collaborative healthcare system model (COHESY)

The second layer is presented by the social network implemented as a web portal which enables different collaboration within the end user community. The social network keeps the data obtained from the mobile application. It consists of various web services that give a restricted access to this data to different end users. The third layer enables interoperability with the primary/secondary health care information systems which can be implemented in the clinical centers and different policy maker institutions. The communication between the first and the second layer is defined by the users' access to the social network where users can store their own data (e.g. personal records, healthcare records, bionetwork records, readings on physical activities). Users can also receive average results from all patients that share the same health condition. Some of these results can be the average levels of certain bio data calculated for certain geographical region, age, sex. Additionally, the data from the social network is used by the recommendation algorithm to recommend physical activities that could be useful to the user.

The communication between the first and the third layer is determined by the communication between the patient and the health care centers. The patient has 24 hour access to medical personnel and a possibility of sending an emergency call. The medical personnel remotely monitors the patient's medical condition, reviews the medical data (e.g. blood pressure, blood-sugar level, heart rate) and responds to the patient by suggesting the most suitable therapy (if different from the one that is encoded in the mobile application) as well as sending him/her various notifications (e.g. tips and suggestions) regarding her/his health condition. The second and the third layer can exchange data and information regarding a larger group of patients grouped by any significant indicator (region, time period, sex, type of the activities) which can be later used for research, policy recommendations and medical campaign suggestions.

COHESY creates an opportunity to increase users' health care within their homes by 24 hour monitoring on one hand, and to increase the medical capacity of health care institutions on the other hand. This results in reducing the overall costs for users and hospitals and improves the user's quality of life. It provides a better health care allowing suggestions and recommendations based on knowledge from other users, cases and experiences. This makes COHESY different from other health care systems. MobiCare [3] and Personal Care Connect [4] are systems that facilitate the remote monitoring of the patients. Both systems consist of a body sensor network, communication infrastructure and servers on which the data is stored. They allow medical personnel and other applications to use the information gathered from sensors. The need for quality of service support in wireless e-health and e-emergency services is discussed by Gama, Carvalho, Alfonso and Mendes in [5]. They emphasize that the network must prioritize the transmission of critical data when sudden change occurs in the patient medical condition, and because of this it is important to distinguish all collected information. Most similar with our system model is Jog Falls system [6]. Jog Falls system is an end to end system to manage diabetes that blends activity and energy expenditure monitoring, diet-logging, and analysis of health data for patients and physicians. This is an integrated system for diabetes management providing the patients with continuous awareness of their diet and exercise, automatic capture of physical activity and energy expenditure, simple interface for food logging, ability to set and monitor goals and reflects on longer term trends.

These examples include research of systems that cover only parts of the presented model. The main difference between the aforementioned examples and Cohesy is the social network, in which the recommendation algorithm that we present in this chapter is implemented.

2. Recommendation algorithm in COHESY

The problem of recommending physical activities cannot be solved using the existing algorithms from collaborative and content-based filtering. The existing algorithms need to have information about the ratings given to items by users. In our case, the items are analogous to physical activities. In other words, we need to know how the physical activities affect change of the value of health parameters and whether that change improves or worsens the health condition of the user. This is another problem that needs to be solved if we want to use the existing recommendation algorithms. Additionally, people that have some diagnosis, for example people with heart problems, should not be recommended some types of activities, for example fast running, because these activities can worsen their health condition. The system should be carefully designed in order to use efficiently all the available information. We propose recommendation algorithm that takes into account all the available information and the complexity of the domain.

For every person and at every moment there is a set of useful activities that can potentially improve her/his health condition. Our recommendation algorithm discovers and recommends these useful activities to the user. The algorithms for activity recommendation must be based on few main principles: (1) Except the physical activities, there are other factors that affect the change of the people's health condition (medicines, food and psychic condition) and the deficiency of this additional information brings to bigger inaccuracy in the recommendations; (2) People can be grouped according to their characteristics (diagnosis, place of living) and for each of these groups activities have specific effect on the parameters which is similar for people in same group, and is different for people in different groups; (3) For every person activities do not influence his/her health with the same intensity and in the same way.

We will separate the health parameters into two groups: descriptive and control parameters. Descriptive parameters express the factors that are important for the people's health (for example age or place of living) and that cannot be affected. Control parameters express the characteristics of the people's health (for example blood pressure or blood sugar level) that can be affected. The goal of the proposed algorithm is to find activities that could change the control parameters towards preferable state which indicates normal health condition.

2.1 Data representation

The main data which will be needed by the recommendation algorithm are:

- The measurements of the health parameters. Each measurement is represented by a vector $Measurement(user, parameter, time, value)$
- The executions of the physical activities. Each execution is represented by a vector $Activity(user, type, time, duration, difficulty)$

For example, if user Alexander runs 5 km in 30 minutes on 29th July, 2013 at 11:00, the $Activity$ vector will look like this:

$$Activity(Alexander, running, 30\ minutes, 29th\ July\ 2013\ 11{:}00, 30\ minutes, 5\ km)$$

Next we give few examples of the Measurement vector:

$$Measurement(Alexander, weight, 28th\ July\ 2013\ 15{:}00, 78\ kg)$$

$$Measurement(Alexander, weight, 30th\ July\ 2013\ 14{:}00, 76\ kg)$$

The difficulty of the activity is a quantitative representation of the given effort to complete that activity. When there is more than one type of activity, the measure of difficulty should be chosen so that if two activities have the same difficulty, the given effort to complete both activities is the same, independently of the types of activity. The measure of difficulty can be analogous to the calories spent to complete the activity.

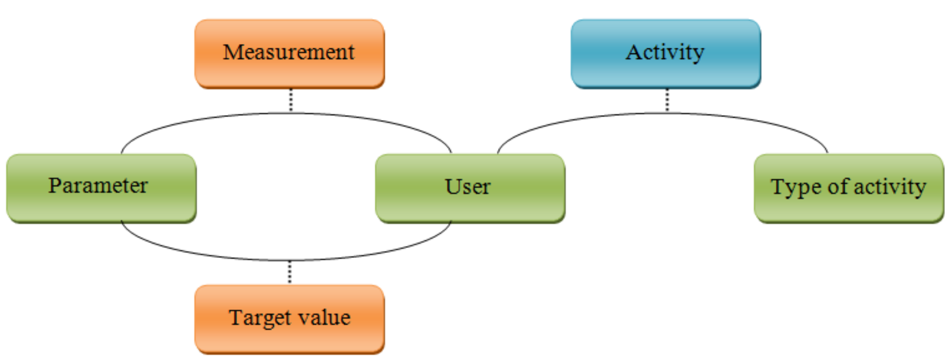

Figure 2. Connections between the different types of data

We will need one more type of data: the users' diagnoses (e.g. heart problems, diabetes). This information will be needed to make the first filtering of users. We can assume that we are given D different types of diagnoses. The vector that describes the diagnoses of some particular user is given by:

$$Restriction(user, diagnosis_1, diagnosis_2 \ \dots \ diagnosis_D)$$

Each $diagnosis_i$ represents a discrete variable that can take one of two values: zero (the user does not have the diagnosis) and one (the user has the diagnosis), or a continuous variable defined over the range [0, 1].

2.2. Data preprocessing

Some of the health parameters have big variance i.e. the measurements performed in close moments differ a lot. These kinds of measurements can be obtained when measuring the blood sugar levels in people that are insulin deficient. The deficiency is caused by the reduced production of insulin in the pancreas [7]. The insulin levels depend on three actions: diet, exercises and insulin injections. The blood sugar levels changes a lot during the day in people that are insulin deficient, so if we are given only two measurements we cannot know for sure whether the health parameter is improved or worsened.

We need to make a transformation of the raw sensor data that will give a more accurate reflection of the health condition of the user at the time of measurement. For that purpose, we can use a function that will transform the readings. This function should be determined by an expert. One function that can be used to find the transformed measurement is:

$$transformedMeasurement(x) = \int_{x-C}^{x} \frac{C-(x-t)}{C} \cdot measurement(t) \cdot dt \qquad (1)$$

where C is the time period before the moment x in which the measurements have an effect on the transformed measurement of the parameter at the moment x. In this function we assume that the older measurements have weaker influence on the final value of the parameter. This formula refers to the ideal case when we are given a continuous function $measurement(t)$ that can be defined mathematically. For more practical application we can use a formula in which the integral is replaced by a weighted sum of all measurements made in the time period C before the moment x multiplied by a factor $\frac{C-(x-t)}{C}$ where t is the moment in which the particular measurement is made. We should note that the result of the transformation of the raw sensor data should represent the parameter that we will try to control through recommendation of physical activities.

Diabetes Data Set [7] contains measurements of the blood sugar level of 70 patients that are diagnosed with diabetes. We chose data from one patient and we applied two transformations on the raw sensor data: weighted average and average from the last 24 hours. We can notice that the transformed measurements are more stable and they have smaller variance.

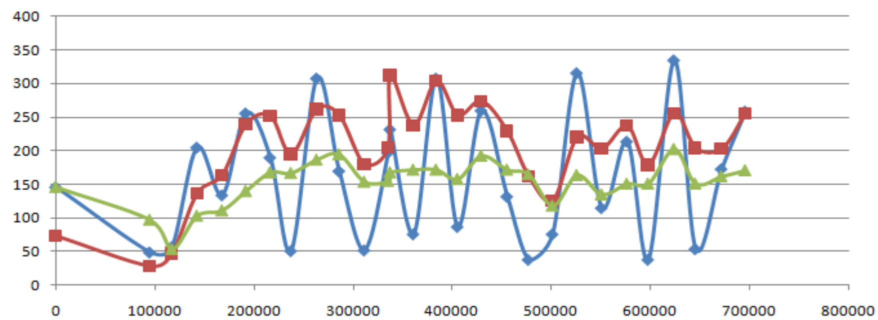

Figure 3. Raw sensor data (blue), weighted average from the last 24 hours (red), average from the last 24 hours (green). The time after the first measurement is represented on the x-axis and the blood sugar level is represented on the y-axis (mg/dl)

2.3 Data fuzzification

In reality, health parameters do not have to be one-dimensional, for example when measuring the blood pressure we obtain two values: systolic and diastolic pressure. In our algorithm each dimension should be treated as a separate health parameter. Another possibility is to map the more-dimensional health parameter into one-dimensional health parameter. Each health parameter has its own characteristics and its value changes on a particular way. We also define a range for each health parameter. Some of the values indicate normal health condition, and some of them indicate worsened health condition. The information about which values are desirable, and which are not, is very important for our recommendation algorithm. That is why we need to map the measurements (optionally transformed) into new classes that will give them semantic meaning. For example, the classes can be: “under normal”, “normal” or “above normal”, or if we are considering the age: “young”, “adult” or “old”. This mapping is made by an expert. On Figure 4 we can see the membership functions for BMI.

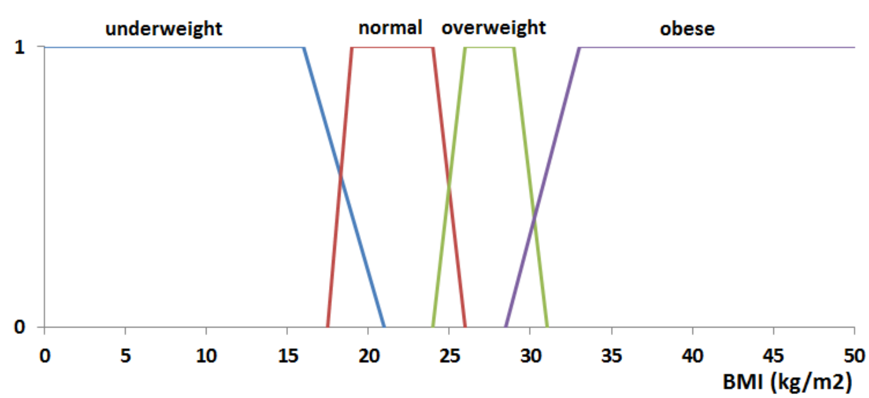

Figure 4. Membership functions for BMI

If we separate the range of the parameter into smaller disjunctive subintervals where each subinterval represents a special class, then we reduce our mapping problem into discretization problem. In this case, if we consider heart rate as a health parameter with normal range [60, 100] heart beats in one minute, two people that have 61 and 59 heart beats in one minute should be placed in different classes. This would not be very useful because the small difference could be because of the noise in the sensor data. Additionally, the two values are on the boundary between two classes and we cannot be fully sure that the first value can be considered as a "normal", and the second value can be considered as "under normal". Each measurement should be assigned a membership to each class. For that purpose we should use fuzzy logic and data fuzzification. This process is also called fuzzy discretization and should be done by an expert.

Let's define the set of membership classes for the parameter *p* with L_p. For each measurement that refers to user *u* we should define the membership $MV_{u,p,l}$ to each parameter class $l \in L_p$. We will denote the membership of the last measurement with $MV_{u,p,l}$, and we will denote the membership of the last measurement before the moment *t* with $MV_{u,p,l,t}$. One measurement belongs to all classes, possibly with different degrees of membership. This means that the mapping of the measurement made at moment *t* can be represented as a vector $FuzzifiedMeasurement$ that consists of all $MV_{u,p,l,t}$ where $l \in L_p$.

2.4 Health profiles

For each user and for each moment there is a health profile that can be defined as a combination of the values of her/his health parameters at that moment. The current health profile is represented by the combination of the most recent parameter readings. Some of the health profiles can be considered as "good" health profiles because the values of the health parameters belong in the normal ranges. Our algorithm tries to arouse transition of the health condition of the user from "bad" health profile to "good" health profile through the physical activities it recommends. Health profile can be represented as a vector:

$$Profile\left(user, time, FuzzifiedMeasurement_1, \dots, FuzzifiedMeasurement_P\right) \tag{2}$$

Health profiles can be generated at regular time intervals (in ideal case after each new measurement). However, we don't need to keep all the profiles in the database, but only the profiles which are considerably different from all health profiles that belong to the user saved so far. In a case where we have new profile $hp_{u,t}$ that belongs to the user *u* at time *t*, first we need to need to calculate the Euclidean distance between the two profiles $ProfilesDistance$ and if the distance between all existing profiles is larger than a given threshold *T*, the new profile is added to the database. The distance metric is defined as:

$$ProfilesDistance\left(hp_{u,t_1}, hp_{v,t_2}\right) = \sum_{p\in P} t_p \cdot MeasurementsDistance_p\left(hp_{u,t_1}, hp_{v,t_2}\right) \tag{3}$$

$$ProfilesDistance\left(hp_u, hp_v\right) = \sum_{p\in P} t_p \cdot MeasurementsDistance_p\left(hp_{u,t_1}, hp_{v,t_2}\right) \tag{4}$$

where

$$MeasurementsDistance_p\left(hp_{u,t_1}, hp_{v,t_2}\right) = \sqrt{\sum_{l\in L_p}\left(MV_{u_i,p,l,t_1} - MV_{u_j,p,l,t_2}\right)^2} \tag{5}$$

$$MeasurementsDistance_p\left(hp_u, hp_v\right) = \sqrt{\sum_{l\in L_p}\left(MV_{u_i,p,l} - MV_{u_j,p,l}\right)^2} \tag{6}$$

In the second formula we use the most recent measurements. We define the significance of the parameter p by t_p. Larger value of t_p means that the measurements of the parameter p will have bigger impact in the calculated distance between two profiles. If we are interested only in improving the state of only one health parameter, then the significance of that parameter should be positive, and the significance of all other parameters should be equal to zero.

2.5 Recommendation algorithm

The algorithm for recommendation of physical activities consists of four main phases:

- Categorization of the users according to their diagnosis and filtering of all users that do not belong to the same category with the active user
- Selection of the users most similar to the active user according to the history of the health profiles by using collaborative filtering
- Calculating the usefulness of the activities to the active user and his similar users by using their health history and history of performed activities
- Generation of recommendations by using the calculated usefulness of the activities

2.5.1 Categorization according to diagnosis

Some physical activities can worsen the health condition of people that have certain diagnosis and that is why they should not be recommended to those people. If the conclusions for the usefulness of the activities are based only on the data from the users belonging to the same group as the active user, they would be more accurate because the activities that have a negative effect to the group of people with a certain diagnosis would get much lower relevance so they would not be recommended. Categorization of the users is made to that users having the same diagnosis and the same set of permissible activities are grouped together. In this phase we need expert knowledge to define the categories and the conditions for membership in these categories.

We need to automate this process because the expert might not be always available. There are two ways to perform categorization: through implementation of expert rules or through building a model using classification algorithms. The first way is more expensive in a case when there are many categories, many different diagnoses or complicated conditions for membership in the categories. That is why the best choice is to use classification algorithms that can build a model using a relatively small set of labeled samples. In this phase of the algorithm we will use the vector $\boldsymbol{Restriction}$ which contains information about the users' diagnoses. First of all, medical expert should define the categories and afterwards she/he should manually classify part of the set of vectors which will be used as a training set. All users should be classified using the classification model. If $\boldsymbol{U}$ is the set of all users and $\boldsymbol{u}^*$ is the active user for which we want to generate recommendations, we should design a function $FilterAccordingToDiagnosis$ that will return the users that belong to the same category as the active user:

$$U^{'} = FilterAccordingToDiagnosis(U, u_*)$$
$$\boldsymbol{U}^{'} \subseteq \boldsymbol{U} \tag{7}$$

If all diagnoses are considered as independent features, we can use Naïve Bayes classifier. We can assume Gaussian distribution for all features and after we determine the parameters of the distributions we will assign the most probable category to each unclassified user.

Decision tree is one kind of inductive learning algorithms that offers an efficient and practical method for generalizing classification rules from previous concrete cases that already solved by domain experts [8]. These kinds of algorithms are useful for many real life applications because the rules are easy to understand. Typically automatically generated diagnostic rules slightly outperformed the diagnostic accuracy of physicians specialists [9]. In our recommendation algorithm we will use decision trees. The most popular decision tree algorithms are IDE, C4.5 and CART [10]. Using these algorithms we can obtain very accurate rules that reflect the expert knowledge used to classify the training set.

The users from the same category have similar health problems. For each category there could be a set of activities that shouldn't be recommended to the users that belong to that category because those activities could worsen their health condition. An expert should determine the set of undesirable activities for all categories. After the user is assigned a category, we should filter out all undesirable activities that refer to that category. In this way we will obtain a set of activities that are potential candidates for recommendation. We define function $FilterAccordingToCategory$ that filters out all activities that could be harmful for the category in which u_* belongs:

$$A^{'} = FilterAccordingToCategory(A, u_*)$$
$$A^{'} \subseteq A \tag{8}$$

In the subsequent phases of the algorithm we should find the usefulness of each type of activity from $A^{'}$ and recommend the most useful activities.

2.5.2 Selection of the most similar users

The second phase of the algorithm is used to select the users that belong to the same class as the active user and that are very similar to the active user regarding their health history. In this way we want to obtain better data that would result with better recommendations. Here we will use the set of health profiles. The main assumption is that if two users had the same combination of parameter values in the past, there is bigger probability that similar latent factors affect their health condition. The current health profile of the active user is analyzed and is compared with the saved health profiles of all other users from the set $U^{'}$. If some user has at least one health profile similar enough (according to some measure such as Euclidean distance) to the current health profile of the active user, then we declare this user as similar to the active user and his data are used in the next phase of the algorithm. For selection we define the function:

$$U^{''} = FilterAccordingToProfile(U^{'}, u_*)$$
$$U^{''} \subseteq U^{'} \tag{9}$$

so that for every $i \in U^{''}$, there is at least one $\hbar p_{u,t}$ for which:

$$ProfilesDistance(\hbar p_{u^*}, \hbar p_{u,t}) < T \tag{10}$$

Data that belongs to users from the set $U^{''}$ (health history and history of performed activities) is used to find the connection between the change of the active user's parameter value and the performed activities. If there are many users that are similar enough to the active user, then we should select k most similar users (for example the first 100). In a case when there are not enough most similar users, the threshold should be increased (in this way there is bigger probability that the generated recommendations will be less relevant).

This type of filtering selects data which could be more useful. If some user in the past had worsened health condition which is the same with the current health condition of the active user and succeeded to improve it on better, her/his data can improve the recommendations for the active user because it potentially contains information about which activities contributed towards improving the health condition. This is on the assumption that in the health history of the similar user there is some improved health condition some period after the moment of storing the profile which is declared as similar to the current health profile of the active user.

2.5.3 Calculating the usefulness of the activities

The main part of the recommendation algorithm is the phase where we calculate the usefulness of the activities. We are given the current health condition of the user (the combination of the values of her/his health parameters) and we want to find activities that would improve the values of the parameters that are into worsened state.

In the previous phase we used both types of health parameters (descriptive and control) to find the most similar users, but in this phase we will use only the set of control parameters $P^{'}$. We examine the history of measurements and performed activities for each user from the set of similar users $U^{''}$. For each performed activity we check its influence on the change of the parameter value. Our main assumption is that the activities that happened in the interval between two measurements influenced the parameter change with intensity that depends on the moment of occurrence and that is proportional to the effort given to complete the activity.

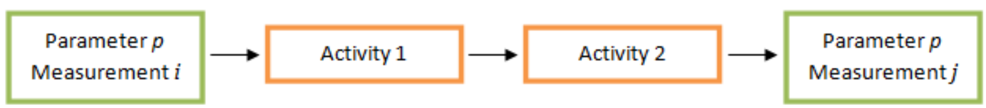

Figure 5. Activities that happened in the interval between two measurements i and j ($i < j$) contributed to the change of the parameter value

We cannot assume that immediately after the completion of some activity there will be a change on the parameter value because some time should pass until the effect of the activity could be visible. On the other side, we cannot assume that the results of a single activity will be visible after a long time period (few months). We create a model that will represent the effect of the activity on the parameter value after its completion. By using this model we find out which measure made after the activity best reflects the effect of that activity. We say that this measurement has the biggest validity. We also need to find out the closest measurement before the activity is started, and its validity can be calculated using the model shown on Figure 6.

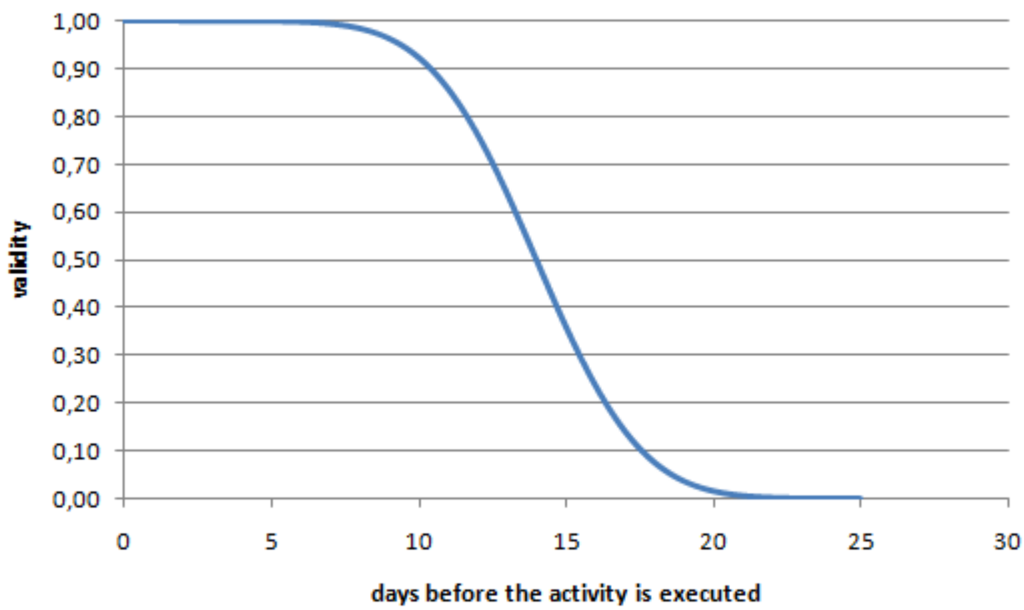

Figure 6. Validity of a measurement before the activity is executed

The function that will define the model should assign bigger validity to measurements that were made shortly before the activity was performed. We need to find the measurement that has the biggest validity and that happened before the activity is executed. On the basis of our requirements for the shape of this validity function, we chose a function that has the form of cumulative normal distribution. Our choice for a function additionally decreases the computational complexity of our algorithm. We also need to find a measurement after the activity is completed that will best reflect the effect of the activity. The validity after the activity should slowly increase, then it should reach a maximum and afterwards it should slowly decrease. For modeling the validity of the measurements after the activity is executed we chose Gamma distribution. This distribution has two parameters and its values are chosen to that the distribution has the form shown on Figure 7. One of the most important characteristics of this validity function is the moment when the maximum is reached. Before and after that moment the validity decreases. In our implementation we choose this moment to be 7 days after the activity is completed. In this way we assume that the maximal influence of the activity is after 7 days. This value can be changed if we are presented with more accurate information.

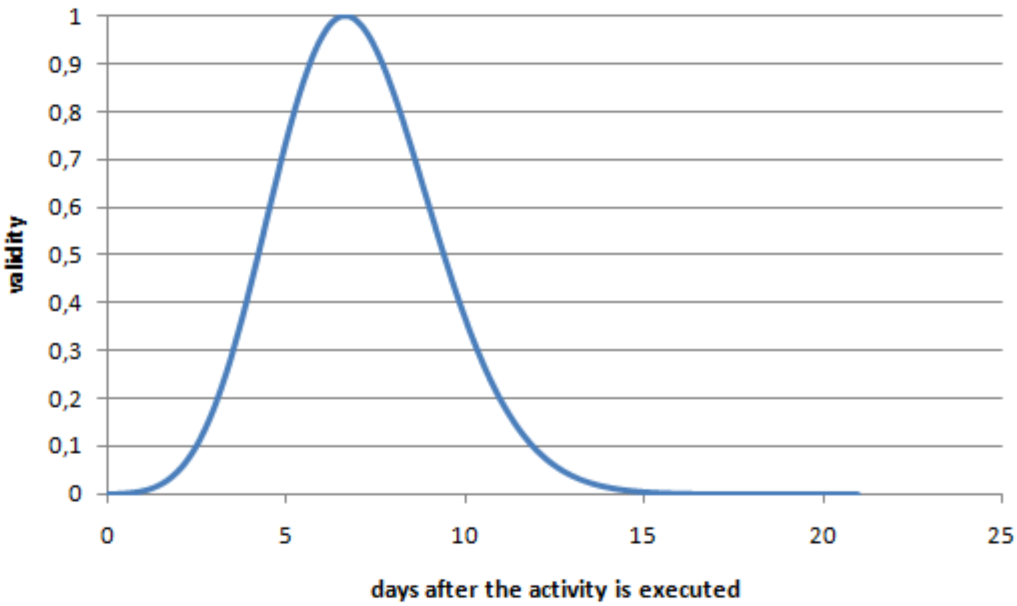

Figure 7. Validity of a measurement after the activity is executed

Let's define a function $dir(u_*, p)$ that as a result returns 1, -1 or 0 depending on whether the value of the parameter p for the active user u_* should be increased, decreased or it shouldn't be changed. The desirable parameter values indicate good health condition and depend on what the active user wants to achieve. By using the validity models (before and after the activity) we define functions $prev_p(activity)$ and $next_p(activity)$ that as a result return the measurements with the biggest validity before and after the activity. We use them in the formula for calculating the usefulness of each type of physical activity to the parameter p for each user u from the set of similar users U'':

$$V_{u,a,p} = \frac{imp_p \cdot \sum_{a_u} \left(\frac{next_p(a_u) - prev_p(a_u)}{timeSpan(duration(a_u))} \right) \cdot val(a_u) \cdot dir(u_*, p) \cdot difficulty(a_u)}{num(a_u)} \qquad (11)$$

$$val(a_u) = validityPrev(a_u) \cdot validityNext(a_u)$$

$$\forall u \in U'', \forall a \in A', \forall p \in P'$$

The variables and the functions that are used in the formula are:

- U'' is the set of similar users for the active user u_*
- A' is the set of different types of activities
- P' is the set of control health parameters
- a_u is a variable that alters all activities of type a performed by user u

- $next_p(a_u)$ is a function that returns the value of the measurement made after activity a_u that has the biggest validity
- $prev_p(a_u)$ is a function that returns the value of the measurement made before the activity a_u that has the biggest validity
- $duration(a_u)$ denotes the duration of the activity a_u
- $difficulty(a_u)$ denotes the difficulty of the activity a_u
- $num(a_u)$ is the number of activities of type a performed by user u
- $val(a_u)$ is a function that returns the validity of the measurements that are used in the analysis of a_u
- imp_p denotes the importance of the parameter in improving the health condition of the active user. The parameters which are into worsened state should have bigger importance. We need to assign even bigger importance to the parameters that are into worsened state and for which it's harder to cause more significant change on their values.
- $timeSpan(x)$ is a function of x where x represents the duration of the activity. This function is an important factor in calculating the change of the parameter value per unit of time. Longer execution of the activity might mean that the user has put smaller effort in completing the activity. The simplest model of this function is $timeSpan(x) = x$. However, in reality we cannot expect that the effect of the activity is disproportionate to the duration of the activity. That is why we suggest this function to have logarithmic form i.e. $timeSpan(x) = \log x$.

In the next step of this algorithm we calculate how useful each type of activity is to user *u*:

$$V_{u,a} = \sum_{p \in P} V_{u,a,p} \tag{12}$$

We insert these values into a matrix of useful types of activities:

$$M = \begin{array}{c} \\ u_1 \\ \vdots \\ u_m \end{array} \begin{array}{c} \begin{array}{ccc} a_1 & \dots & a_r \end{array} \\ \begin{bmatrix} V_{1,1} & \cdots & V_{1,r} \\ \vdots & \ddots & \vdots \\ V_{m,1} & \cdots & V_{m,r} \end{bmatrix} \end{array} \tag{13}$$

2.5.4 Generation of recommendations

There are few different ways to utilize the results produced in the previous phase in order to make recommendations. The simplest method is to calculate the usefulness V_a for each type of activity according to the formula:

$$V_a = \sum_{u \in U''} V_{u,a} \tag{14}$$

and to recommend the activity with the maximal V_a. This is not always practical because some type of activity could have significantly bigger usefulness to some user (comparing with the other users) and that value will dominate in the sum. In this way a type of activity that is very useful only to a small subset of the set of similar users would be recommended to the active user. Another way is to recommend the activity which is considered as the most useful to the largest number of similar users. First, for each user from the set of similar users we should create a vector of useful activities:

$$\forall u \in U^{''}, \quad UA_u = k \quad where \quad V_{i,k} = \max_{j \in A^{'}} V_{i,j}$$
$$UA = [UA_1 \dots UA_m] \tag{15}$$

$$\forall k \in A^{'} \quad count(k) = \sum_{i=1}^{m} r_i \quad \text{where} \quad r_i = \begin{cases} 1, k = UA_i \\ 0, k \neq UA_i \end{cases}$$
$$RA = u \quad where \quad count(u) = \max_{k \in A^{'}}(count(k)) \tag{16}$$

Activity type RA is recommended to the active user u_*.

3. Evaluation by simulation

Our recommendation algorithm tries to find the usefulness of each type of activity on the bio-medical parameter change. Activity is considered useful if it changes the global parameter value towards the desired one. The change of a parameter value might be influenced by many factors. It is impossible to make a mathematical model that takes into account all these factors, so we tried to make a model for the parameter change, under the influence of the activities performed, that is simple and as closer to the reality as possible. We assume that each performed activity has some influence on the parameter change and that the parameter change is influenced only by the effect of the activities (pharmacological influence is neglected).

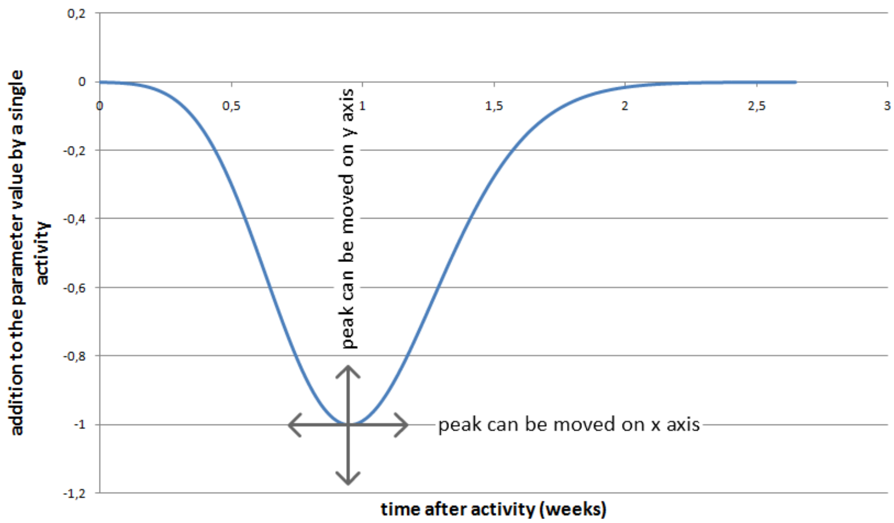

Figure 8. Influence on the global parameter change by a single activity

We model a single activity influence to the global parameter change by a function whose shape is similar to a Poisson probability mass function (Fig. 8). In our model, it depends on the type of activity. The peak of the model function can have either positive or negative amplitude depending on whether it has a stimulatory or inhibitory effect correspondingly. Gaussian noise is added to the model function. When multiple activities from different types are performed, each of them influences the parameter value. So, the parameter value at each moment of time can be defined by the expression:

$$f(t) = \sum_i h(type_i, timeOfStart_i, t) \quad (17)$$

Gaussian noise is also added to $f(t)$. On Figure 9 and Figure 10 we can observe two global parameter functions, the first one without noise and the second one with noise.

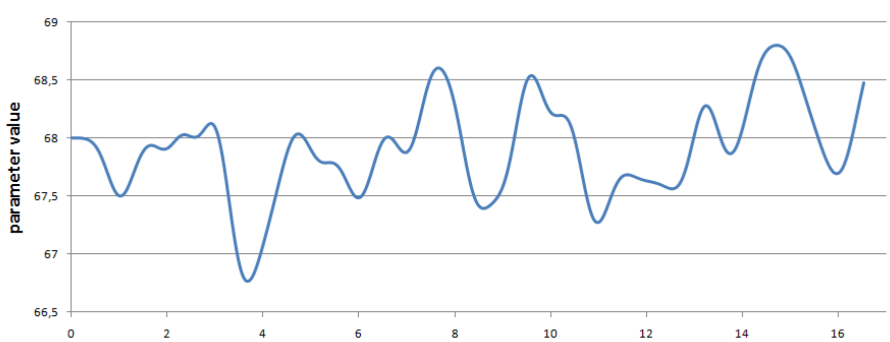

Figure 9. Global parameter function without noise

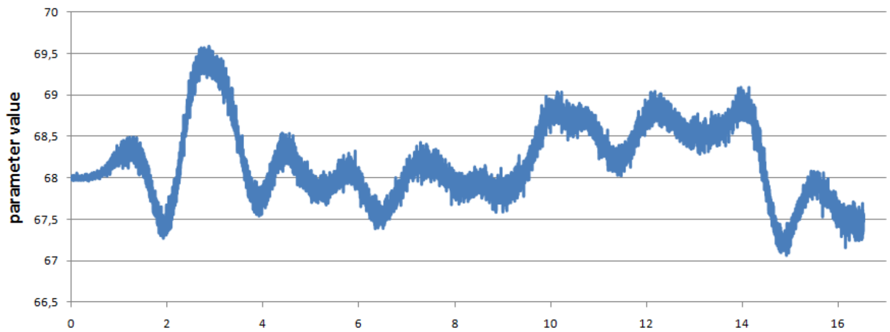

Figure 10. Global parameter function with noise

The simplest case to test the correctness of our algorithm is to use two types of activities which are symmetrical. The first one has positive peak amplitude and the second one has negative peak amplitude. Our algorithm should "guess" which activity increases the parameter value and which doesn't. If we flipped a coin we could guess with 50% accuracy which activity has more usefulness (in our experiment we assume that activity is useful if it increases the global parameter value). In our simulator we use these parameters:

- Duration of the simulation
- Average time between consecutive activities
- Average time between consecutive measurements
- X coordinate of the peak (relative to the start of the activity and same for both activities)
- Y coordinate of the peak (both activities have peaks symmetric about the y-axis)
- Standard deviation of the peak (x coordinate)
- Standard deviation of the peak (y coordinate)
- Standard deviation of the Gaussian noise (single influence)
- Standard deviation of the Gaussian noise (global parameter function)

We expected that by increasing the standard deviation we would get lower accuracy, but by increasing the duration of the simulation and the average time between consecutive activities we would get higher accuracy. Longer duration of the simulation means more activities and more data and longer average time between consecutive activities means that it could be easier to distinguish between consecutive activities by observing the global parameter function. On the other side, longer average time between consecutive measurements means that we will have less data for our recommendation algorithm and less accurate recommendations. We applied our recommendation on generated data and we obtained correct results when we didn't have standard deviation of the peak and of the noise. We wanted to know how the algorithm behaves when we change the parameters of the simulation. We were especially interested how the accuracy of the recommendation algorithm changes in the border cases (Figure 11 and Figure 12).

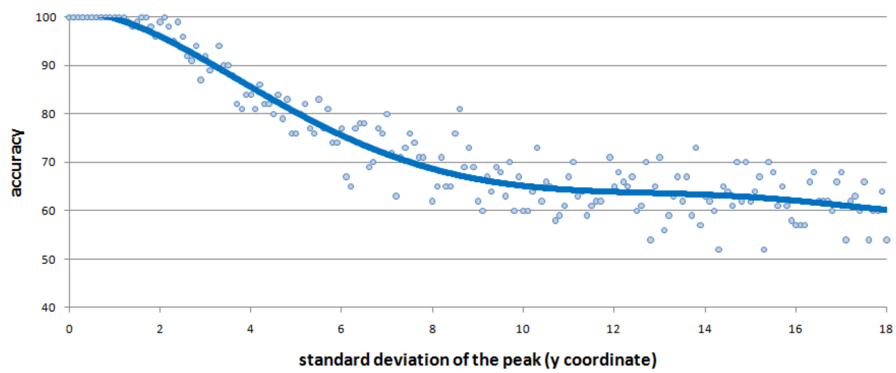

Figure 11. Accuracy as a function of the standard deviation of the peak (y coordinate)

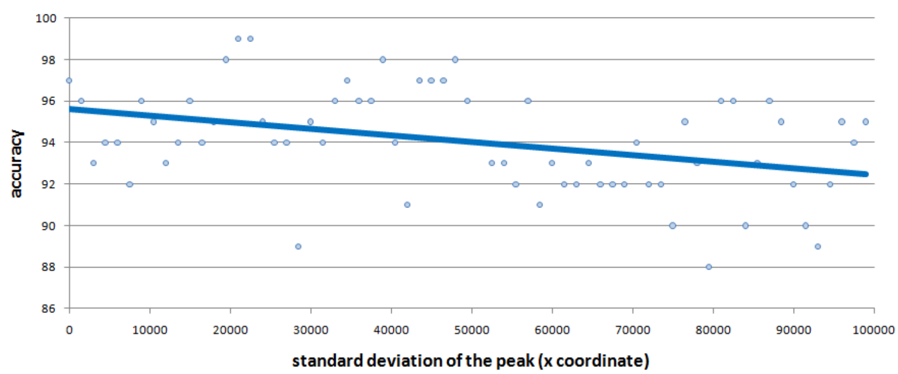

Figure 12. Accuracy as a function of the standard deviation of the peak (x coordinate)

We chose a set of values for the parameters of the simulator in order to see more clearly the way the accuracy changes. We tried to define a curve that fits the results we obtained. In the first case, when we observed the accuracy as a function of the standard deviation of the peak (y coordinate) we noticed that there is some threshold after which the algorithm gives bad results, however, this threshold is relatively big comparing to the amplitude of the peak. Surprisingly, the increasing of the standard deviation of the peak (x coordinate) didn't cause significant worsening of the results.

The increase of standard deviation of the noise caused exponential decline of the accuracy. This is shown on Fig. 13 and Fig. 14.

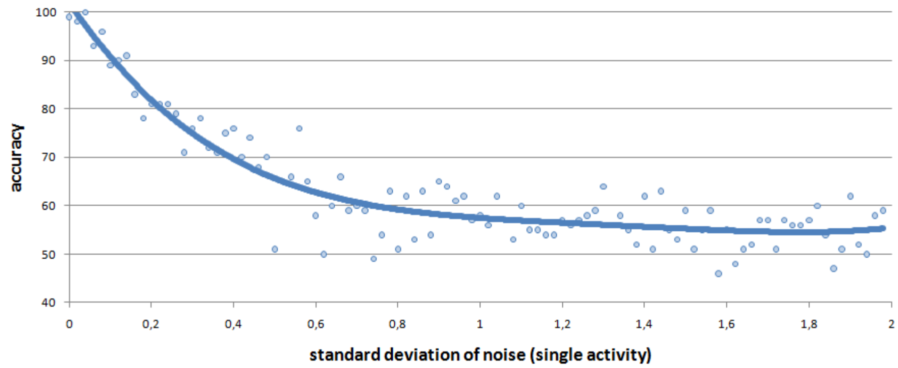

Figure 13. Accuracy as a function of the standard deviation of the noise (single activity)

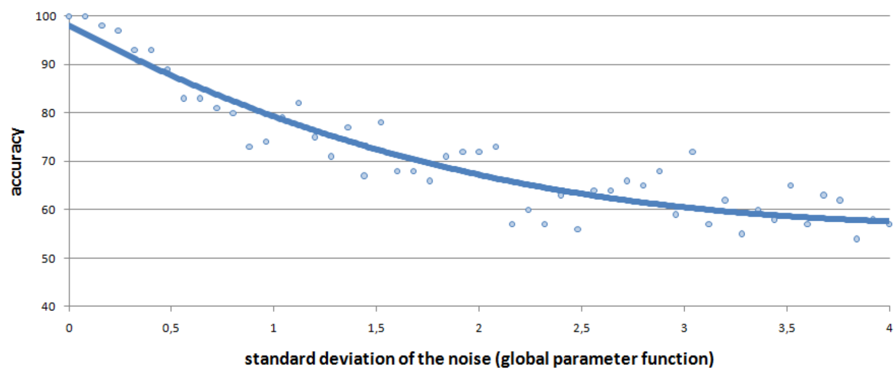

Figure 14. Accuracy as a function of the standard deviation of the noise (global parameter function)

Case studies

Case study 1

The user (with heart problems) switches on the application on his phone. The application connects to Bluetooth devices which measure his heart rate, blood pressure and blood sugar level. The application uses its knowledge base to determine whether his health parameters are in the normal ranges. The application searches for users that have the same diagnosis and it concludes that walking for one hour could improve her/his health condition. The user tells the application that she/he will start her/his activity (walking). The application tracks the GPS coordinates at every moment during the activity. In the end of the activity the user tells the application that she/he has finished the activity.

Case study 2

Some person A decides to practice more cycling. She/he uses the mobile application to record data each day about her/his weight, blood pressure, blood-sugar level and heart rate. Every two days she/he cycles for about 1 hour. She/he starts the application before she/he starts with the activity. The application measures the distance passed and the velocity. After the activity, the application decides what the level of difficulty of the activity was. All these data are saved in the social network. After 2 months the user notices that she/he has lost weight (3 kilograms). After that some other user B installs the application on her/his mobile phone and tries to get recommendations for losing weight. First she/he records data about her/his health parameters and she/he tells the application that she/he wants to lose weight. The application finds other users which are similar to the user B (gender, weight, heart rate, diagnosis). Among them is the user A. Some of these users succeeded to lose weight in the past. One of them is user A. The application concludes that cycling is the best activity to lose weight and this activity is recommended to the user A.

Future directions

The future healthcare systems will have sense making component in order to increase the understanding of patient's personal condition, clinical or living environment and, in that way, to increase the healthcare system efficiency by increasing its availability. It is very important to discuss the ways in which information systems that support healthcare can make that kind of sense making. COHESY is a health care system that gathers a lot of information from the patient's bionetwork, his medical record, the environment, his interaction with the mobile application, and this information is used to make conclusions about the health condition of the patient and the ways in which it can be improved. Its recommendation algorithm gives the people bigger control over their own health and supports their actions. We expect that besides the presented algorithm, other decision support algorithms will appear that will utilize the data provided by COHESY. They could focus on making different types of conclusions. The increase of the number of users will bring more data and this means that the generated conclusions and recommendations could be more accurate. The patterns that are hidden behind the connections between patients can lead to better understanding of their current health condition. That is why we believe that collaborative algorithms could be very beneficial for health care systems.

There are few other reasons why we can expect that collaborative health care systems will be intensively used in the future. The first reason is that the technology is advancing rapidly and this means that the sensors would become more accurate and would be easier to use. We can also expect that new sensors would emerge that could measure different health parameters. This leads to bigger and better data available to the collaborative algorithms. The second reason is that the recommendation systems and collaborative systems could be a promising approach for preventive and curative healthcare solutions. The development of these algorithms along with additional information which can be relevant in health care can lead to more accurate recommendations.

Modern medical technology helped a lot of people, and there will be even deeper integration of the technology in the health care systems. These systems will be more aware of the health condition of the patients and the types of conclusions will be even closer to the conclusions made by doctors. Our proposed system is a step towards smarter health care. The recommendations help people to be more aware of their actions and the consequences that they cause.

Conclusion

The recommendation algorithm presented in this chapter is the main component of the collaborative health care system model – COHESY. The purpose of the algorithm is to find the dependency of the users' health condition and the physical activities she/he performs. It gives recommendations for performing specific activities that would improve users' health. To achieve this we consider datasets from the history of measurements and the history of performed activities of users. Data is obtained by the user through its interaction with the mobile application and by sensors connected to the smartphone.

The algorithm consists of four phases: categorization of the users according to their diagnosis, selection of the users most similar to the active user, calculating the usefulness of the activities and generation of recommendations. First two steps are not necessary, but they contribute to improving the quality of the recommendations by selecting the most relevant data. We assume that if two users had the same combination of parameter values in the past, there is bigger probability that similar latent factors affect their health condition. The selection of the similar users and the usage of their data represent the collaborative component of our algorithm.

The proposed algorithm is different from the standard recommendation algorithms because of the complexity of the problem it solves. The time dimension and the uncertainty of the feedback make the solving of the problem difficult. We want to find the influence of the physical activities on the change of the health condition, but there are many other factors that affect on this change, for example: medications, food, and emotional state. That is why we cannot assume that the generated recommendations will be always correct. We try to overcome this downside by increasing the amount of data used for calculating the usefulness of the activities.

The fuzziness which is included in the processing of the sensor data allows us to obtain their semantic meaning. In this way the algorithm receives information about which parameter values indicate normal state, and which indicate worsened state. Also, this representation allows the algorithm to have equal look at the health parameters regardless of their characteristics and the different ranges in which their values belong.

The experiments conducted with generic data show that the accuracy of the algorithm increases with the duration of the observation i.e. with the number of activities observed and the amount of data obtained. The behavior of the algorithm on the increased data uncertainty is also evaluated. The results show that the algorithm is very robust and promising. It can be implemented efficiently and it offers a lot of possibilities for adaptation. The generated recommendations allow the user to adapt and align her/his physical activities in order to improve her/his health condition with bigger confidence. In this way, users have bigger control on their own health.

There are few reasons why we can expect that collaborative health care systems will be intensively used in the future. The first reason is that the technology is advancing rapidly and this means that the sensors would become more accurate and would be easier to use. We can also expect that new sensors would emerge that could measure different health parameters. The second reason is that the recommendation systems and collaborative systems could be a promising approach for preventive and curative healthcare solutions. The development of these algorithms along with additional information which can be relevant in health care can lead to more accurate recommendations.

References

1. Vlahu-Gjorgievska, E., Trajkovik, V.: Towards collaborative health care system model – COHESY. In: Proceedings of the 2011 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), pp. 1-6. IEEE (2011)
2. Taylor, C., Dajani, L.: The future of homecare systems in the context of the ubiquitous web and its related mobile technologies. In: Proceedings of the 1st International conference on Pervasive Technologies Related to Assistive Environments. P 43. ACM (2008)
3. Chakravorty, R.: A programmable service architecture for mobile medical care. In: Proceedings of UbiCare 2006: The first workshop on Ubiquitous and Pervasive Healthcare. pp. 5. IEEE (2006)
4. Blount, M., Batra, V.M., Capella, A.N., Ebling, M.R., Jerome, W.F., Martin, S.M., Nidd, M., Niemi, M.R., Wright, S.P.: Remote health-care monitoring using personal care connect. IBM Systems Journal 46(1), 95-113 (2007)
5. Gama, O., Carvalho, P., Afonso, J.A., Mendes, P.: Quality of service support in wireless sensor networks for emergency healthcare services. In: Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE. pp. 1296-1299. IEEE (2008)
6. Nachman, L., Baxi, A., Bhattacharya, S., Darera, V., Deshpande, P., Kodalapura, N., Mageshkumar, V., Rath, S., Shahabdeen, J., Acharya, R.: Jog falls: a pervasive healthcare platform for diabetes management. In: Pervasive Computing, pp. 94-111. Springer (2010)
7. Kahn, M. (1994). Diabetes Data Set. http://archive.ics.uci.edu/ml/datasets/Diabetes.
8. Chao, S., Wong, F.: An incremental decision tree learning methodology regarding attributes in medical data mining. In: Machine Learning and Cybernetics, 2009 International Conference on. vol. 3, pp. 1694-1699. IEEE (2009)
9. Kononenko, I.: Inductive and Bayesian learning in medical diagnosis. Applied Artificial Intelligence: An International Journal, 7(4), 317-337 (1993)
10. D. Lavanya, Rani, D. K.: Performance evaluation of decision tree classiffers on medical datasets. International Journal of Computer Applications 26(4), 1-4 (2011)

Keywords

Personal healthcare, recommendation algorithm, fuzzy-discretization, classification, data preprocessing, distance metric, mathematical model, evaluation.

Questions and Answers

1. How many usage levels does COHESY have? Please describe each of them.

   COHESY is deployed over three basic usage layers. The first layer consists of the bionetwork (implemented from various body sensors) and a mobile application that reads sensor data and allows the users to specify the activities they perform. The second layer is presented by the social network implemented as a web portal which enables different collaboration within the end user community. The third layer enables interoperability with the primary/secondary health care information systems which can be implemented in the clinical centers and different policy maker institutions.

2. How do we separate the health parameters?

   We separate the health parameters into two groups: descriptive and control parameters. Descriptive parameters express the factors that are important for the people's health (for example age or place of living) and that cannot be affected. Control parameters express the characteristics of the people's health (for example blood pressure or blood sugar level) that can be affected. The algorithm tries to change the only the control parameters towards desirable state, but both types of parameters are used in the phase of the algorithm where we select the users most similar to the active user.

3. What types of recommendations are generated by the recommendation algorithm in COHESY?

   The recommendation algorithm in COHESY gives recommendations about physical activities that should be performed by the patients in order to improve their health condition.

4. What types of data are used by the recommendation algorithm?

   The recommendation algorithm uses the history of performed activities and the history of measurements of all users in order to generate recommendations.

5. What is a health profile?

   The combination of the user's values of her/his health parameters represents his health profile.

6. Which are the four main phases of the recommendation algorithm in COHESY?

   The algorithm consists of four phases: categorization of the users according to their diagnosis, selection of the users most similar to the active user, calculating the usefulness of the activities and generation of recommendations.

7. Which phases can be omitted by the recommendation algorithm, but are important to generate more accurate recommendations?

   First two phases are not necessary, but they contribute to improving the quality of the recommendations by selecting the most relevant data.

8. How do we choose which measurements made before and after a particular activity are most relevant to describe the effect of that activity on the health parameter?

   We chose the most recent measurement made before the activity and we choose the measurement made after the activity with has the biggest validity (relevance) according to a validity function. We cannot assume that immediately after the completion of some activity there will be a change on the parameter value because some time should pass until the effect of the activity could be visible. On the other side, we cannot assume that the results of a single activity will be visible after a long time period (few months). That is why this validity function gives smaller relevance to measurements made immediately after the activity and to measurements made very long time after the activity.